REMARKS

In the Official Action mailed on **1 April 2007**, the Examiner reviewed claims 1-20. The specification was objected to because of informalities. Claims 19 and 20 were rejected under 35 U.S.C. § 101 because the claimed invention is directed to non-statutory subject matter. Because Examiner did not address Applicant's amendments to overcome objections to the specification and rejections under 35 U.S.C. § 101, and because Examiner does not maintain these objections in the present Office Action, Applicant assumes that Applicant's prior amendments were sufficient to overcome these objections.

In the Official Action mailed on **01 October 2007**, the Examiner reviewed claims 1-20. Claims 1-7, 9-16 and 18-20 were rejected under 35 U.S.C. § 102(e) as being anticipated by Rajwar et al. (USPN 7,120,762, hereinafter "Rajwar"). Claims 8 and 17 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Rajwar, in view of Hecht et al. (U.S. Pub. No. 2003/0064808, hereinafter "Hecht").

Rejections under 35 U.S.C. § 102(e)

Examiner rejected claims 1-20 under 35 U.S.C. § 102(e), asserting that these claims are anticipated by Rajwar. More specifically, Examiner argues that the Rajwar discloses "atomically committing changes made during transactional execution" (see Office Action page 3, sec. 6). Applicant respectfully disagrees because Rajwar discloses only freely buffering results into an L1 cache and then updating the L2 cache with the results in the L1 cache, whereas atomically committing changes made during transactional execution involves (among other operations, as described below) protecting cache lines modified during transactional execution from interfering accesses while the results from transactional execution are written from a store buffer to the cache lines.

The Rajwar system monitors code execution to detect the beginning of a critical section. When critical section is executed, the Rajwar system elides the lock-acquiring and lock-releasing steps and speculatively executes the critical section (see Rajwar, claim 1 and col. 2, lines 30-33). At the end of the speculative-execution, if there has been no "interruption" to the access of the portion of a common memory used by the system during the speculative execution, the system "commits the speculative execution of the critical section" (see Rajwar, claim 1 and col. 9, lines 46-49). As described in Rajwar, committing the speculative execution involves only "**updating cache L2 with the L1 cache**." More specifically, Rajwar explicitly discloses using the L1 cache as a "buffer" for the speculative execution of the critical section in order to prevent the effects of the critical section from being observed by other processor units and **writing the speculative results from the L1 cache to the L2 cache at the end of the speculative execution** (see Rajwar, col. 8, lines 35-40). Rajwar discloses only "*standard cache protocol messages*" being used to detect conflicts (see Rajwar, col. 8, lines 48-50). Nothing in Rajwar discloses atomically committing changes made during the transactional execution.

In contrast, in embodiments of the present invention, **the system atomically commits the results from the transactional execution**. More specifically, the system starts by **treating cache lines as though they are locked** (see instant application, par. [0076]-[0079]). This means other processes that need to access a cache line must wait until the line is no longer locked before they can access the cache line (i.e., similar to how lines are locked in conventional caches). Next, the system clears marks from the cache lines in the L1 data cache. The system then commits entries from the store buffer for stores that were generated during transactional execution. As each entry is committed, the corresponding cache line is unlocked. (The system also commits register file changes using a flash copy.) *Atomically committing the results of the transaction*

permits processes to share all levels of the cache hierarchy during a transaction (i.e., processes can share the L1).

Moreover, the sections cited by Examiner in support of the rejection using Rajwar do not disclose the indicated subject matter. Specifically:

- col. 3, lines 15-17: describe continuing non-transactional execution at the conclusion of speculative execution;
- col. 5, lines 57-60: generally describe atomic instructions (with no discussion of speculative execution that involves using atomic instructions to complete speculative execution); and
- col. 9, lines 45-50: describe updating the L2 cache from the L1 cache.

Applicant respectfully avers that none of these sections (or any other section of Rajwar) are proper support for the rejection under 35 U.S.C. § 102.

In summary, the Rajwar system depends on being able to freely write the results from transactional execution into the L1 cache (or else return to a conventional technique of getting locks for the data structures) and then write those results to the L2 cache at the end of the speculative execution. Nothing in Rajwar discloses atomically committing changes made during the transactional execution.

Accordingly, Applicant has amended independent claims 1, 9, and 19 to clarify that embodiments of the present invention atomically commit the results from the transactional execution. Support for these amendments can be found in paragraph [0075]-[0079] of the instant application. No new material has been added.

Hence, Applicant respectfully submits that independent claims 1, 9, and 19 are in condition for allowance. Applicant also submits that claims 2-9, which depend upon claim 1, claims 11-18, which depend upon claim 10, and claim 20, which depends upon claim 19, are in condition for allowance and for reasons of the unique combinations recited in such claims.

CONCLUSION

It is submitted that the present application is presently in form for allowance. Such action is respectfully requested.

Respectfully submitted,

By __/Anthony Jones/_____
Anthony Jones
Registration No. 59,521

Date: 3 December 2007

Anthony Jones
Park, Vaughan & Fleming LLP
2820 Fifth Street
Davis, CA 95618-7759
Tel: (530) 759-1666
Fax: (530) 759-1665
Email: tony@parklegal.com